In the Claims:

1. (Currently Amended) A server load reduction system for viewing data from a master URL comprising:

a client group of computers comprising a plurality of browsers receiving the master URL to browse to only when the data representing a target page has been completely distributed to said client group of computers;

a multicast server client storage location comprising a client browser cache and comprising logic automatically distributing the data to said client group of computers via multicast file distribution; and

[[a]]at least one client server for determining that a potential URL is the desired master URL and loading the master URL to said multicast server client storage location.

2. (Canceled)

3. (Previously Presented) A server load reduction system according to claim 1 wherein at least two members of said client group of computers operate different web browser programs.

4. (Canceled)

5. (Previously Presented A server load reduction system according to claim 1 wherein the data is transferred to said client server from a proxy server through a SERGE transport system.

6. (Canceled)

7. (Previously Presented) A method for reduction of server load comprising:

conducting a browse operation with a proxy browser to find a master URL;

requesting data contained in said master URL for use by a plurality of client

computers;

receiving said data in a client server;

storing said data in a client browser cache;

automatically loading said data to said plurality of client computers from said client browser cache;

said plurality of client computers receiving said master URL to browse to only when all of said data is loaded to said plurality of client computers; and

said plurality of client computers attempting to browse to said master URL, whereby said plurality of client computers load said data on monitors of said plurality of client computers.

8-9. (Canceled)

10. (Currently Amended) A method according to claim 7 further comprising the step of updating said client browser cache to ~~contain substantially~~continuously maintain current said master URL data.

11. (Currently Amended) A method for reduction of server load comprising:

conducting a browse operation with a proxy browser to find a master URL;

requesting a unicast portion of data contained in said master URL for use by a first client;

receiving said unicast portion of said data in a proxy server;

storing said unicast portion of said data in said proxy server;

notifying a first client server when a proxy server contains all of said unicast portion of said data;

determining that said master URL is a desired master URL;

requesting a multicast portion of said data contained in said master URL for use by said first client;

receiving said multicast portion of said data in said proxy server;

notifying a multicast client server when said proxy server contains all of said

multicast portion of said data;

receiving said multicast portion of said data in said multicast server;

automatically loading said multicast portion of said data from said multicast server to a plurality of client computers;

said plurality of client computers receiving said master URL to browse to only when all of said data is loaded to said plurality of client computers; and

said plurality of client computers attempting to browse to said master URL, whereby said plurality of client computers load said data on monitors of said plurality of client computers.

12. (Original) A method according to claim 11 wherein requesting said unicast portion of said data contained in said master URL further comprises requesting said unicast portion of said data contained in said master URL for use by a second client.

13. (Original) A method according to claim 12 wherein requesting said multicast portion of said data contained in said master URL further comprises requesting said multicast portion of said data contained in said master URL for use by said second client.

14. (Previously Presented) A method according to claim 12 further comprising downloading said multicast portion of said data to said first client server.

15. (Previously Presented) A method according to claim 12 wherein notifying said first client server when said proxy server contains all of said unicast portion of said data further comprises notifying said second client server when said proxy server contains all of said unicast portion of said data.

16. (Previously Presented) A method according to claim 12 further comprising downloading said multicast portion of said data to said second client

server.

17. (Currently Amended) A method according to claim 11 further comprising the step of updating said proxy server to ~~contain substantially~~maintain current master URL data.

18. (New) A server load reduction system according to claim 1 wherein said at least one client server comprises:

a proxy server downloading said data;

a first multicast server coupled to said client server and receiving said data from said client server and generating a signal indicative of all said data being transferred to said first multicast server; and

a second multicast server coupled to said first multicast server and storing said data in said multicast server client storage location.

19. (New) A server load reduction system according to claim 18 wherein said second multicast server loads said client group of computers with portions of said data as desired by said client group of computers.

20. (New) A server load reduction system according to claim 18 further comprising a unicast associated client server coupled to said proxy server and transmitting unicast associated data to said client group of computers.

21. (New) A method according to claim 7 further comprising:

signaling a proxy server cache when a portion of said master URL is modified;

reloading said master URL data ; and

distributing said master URL data to said client server.